BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

As rescanning documents *will not* correct images, please do not report the images to the Image Problem Mailbox.

Search Results

Search Results for: **[data mining <and> conditional probability]**
Found **89** of **121,059 searched.**

Search within Results

> Advanced Search

> Search Help/Tips

Sort by: Title Publication Publication Date Score Binder

Results 21 - 40 of 89 short listing

Prev Page 1 2 3 4 5 Next Page

21 Interactive path analysis of web site traffic 80%
Pavel Berkhin , Jonathan D. Beche , Dee Jay Randall
Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining August 2001
The goal of ***Path Analysis*** is to understand visitors' navigation of a Web site. The fundamental analysis component is a path. A path is a finite sequence of elements, typically representing URLs or groups of URLs. A full path is an abstraction of a visit or a session, which can contain attributes described below. Subpaths represent interesting subsequences of the full paths.Path Analysis provides user-configurable extraction, filtering, preprocessing, noise reduction, descriptive st ...

22 Discovering critical edge sequences in E-commerce catalogs 80%
Kaushik Dutta , Debra VanderMeer , Anindya Datta , Krithi Ramamritham
Proceedings of the 3rd ACM conference on Electronic Commerce October 2001
Web sites allow the collection of vast amounts of *navigational data* -- clickstreams of user traversals through the site. These massive data stores offer the tantalizing possibility of uncovering interesting patterns within the dataset. For e-businesses, always looking for an edge in the hyper-competitive online marketplace, this possibility is of particular interest. Of significant particular interest to e-businesses is the discovery of *Critical Edge Sequences* (CES), which denote f ...

23 Text categorization for multi-page documents: a hybrid naive Bayes HMM approach 80%
Paolo Frasconi , Giovanni Soda , Alessandro Vullo
Proceedings of the first ACM/IEEE-CS joint conference on Digital libraries January 2001
Text categorization is typically formulated as a concept learning prob lem where each instance is a single isolated document. In this paper we are interested in a more general formulation where

documents are organized as page sequences, as naturally occurring in digital libraries of scanned books and magazines. We describe a method for classifying pages of sequential OCR text documents into one of several assigned categories and suggest that taking into account contextual information provid ...

24 Information dependencies 80%
Mehmet M. Dalkilic , Edward L. Roberston
Proceedings of the nineteenth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems May 2000
This paper uses the tools of information theory to examine and reason about the information content of the attributes within a relation instance. For two sets of attributes X and Y, an information dependency measure (InD measure) characterizes the uncertainty remaining about the values for the set Y when the values for the set X are known. A variety of arithmetic inequalities (InD inequalities

25 Discovering roll-up dependencies 80%
Jef Wijsen , Raymond T. Ng , Toon Calders
Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining August 1999

26 Powerful image organization in visual retrieval systems 80%
Marinette Bouet , Chabane Djeraba
Proceedings of the sixth ACM international conference on Multimedia September 1998

27 Beyond market baskets: generalizing association rules to correlations 80%
Sergey Brin , Rajeev Motwani , Craig Silverstein
ACM SIGMOD Record , Proceedings of the 1997 ACM SIGMOD international conference on Management of data June 1997
Volume 26 Issue 2
One of the most well-studied problems in data mining is mining for association rules in market basket data. Association rules, whose significance is measured via support and confidence, are intended to identify rules of the type, “A customer purchasing item A often also purchases item B.” Motivated by the goal of generalizing beyond market baskets and the association rules used with them, we develop the notion of mining rules that identify correlations (generalizing associations ...

28 Posters: Query word deletion prediction 77%
Rosie Jones , Daniel C. Fain
Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval July 2003
Web search query logs contain traces of users' search modifications. One strategy users employ is deleting terms, presumably to obtain greater coverage. It is useful to model and automate term deletion when arbitrary searches are conjunctively matched against a small hand constructed collection, such as a hand-built hierarchy, or collection of high-quality pages matched with key phrases. Queries with no matches can have words deleted till a match is obtained. We provide algorithms which perform ...

29 Text categorization: Robustness of regularized linear classification methods in text categorization 77%

Jian Zhang , Yiming Yang

Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval July 2003

Real-world applications often require the classification of documents under situations of small number of features, mis-labeled documents and rare positive examples. This paper investigates the robustness of three regularized linear classification methods (SVM, ridge regression and logistic regression) under above situations. We compare these methods in terms of their loss functions and score distributions, and establish the connection between their optimization problems and generalization error ...

30 Web: Implicit link analysis for small web search 77%

Gui-Rong Xue , Hua-Jun Zeng , Zheng Chen , Wei-Ying Ma , Hong-Jiang Zhang , Chao-Jun Lu

Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval July 2003

Current Web search engines generally impose link analysis-based re-ranking on web-page retrieval. However, the same techniques, when applied directly to small web search such as intranet and site search, cannot achieve the same performance because their link structures are different from the global Web. In this paper, we propose an approach to constructing implicit links by mining users' access patterns, and then apply a modified PageRank algorithm to re-rank web-pages for small web search. Our ...

31 Survey articles: Data mining for hypertext: a tutorial survey 77%

Soumen Chakrabarti

ACM SIGKDD Explorations Newsletter January 2000

Volume 1 Issue 2

With over 800 million pages covering most areas of human endeavor, the World-wide Web is a fertile ground for data mining research to make a difference to the effectiveness of information search. Today, Web surfers access the Web through two dominant interfaces: clicking on hyperlinks and searching via keyword queries. This process is often tentative and unsatisfactory. Better support is needed for expressing one's information need and dealing with a search result in more structured ways than av ...

32 A survey of data mining and knowledge discovery software tools 77%

Michael Goebel , Le Gruenwald

ACM SIGKDD Explorations Newsletter June 1999

Volume 1 Issue 1

Knowledge discovery in databases is a rapidly growing field, whose development is driven by strong research interests as well as urgent practical, social, and economical needs. While the last few years knowledge discovery tools have been used mainly in research environments, sophisticated software products are now rapidly emerging. In this paper, we provide an overview of common knowledge discovery tasks and approaches to solve these tasks. We propose a feature classification scheme that can be ...

33 Securing information: Guarding the next Internet frontier: countering denial of information attacks 77%

Mustaque Ahamad , Leo Mark , Wenke Lee , Edward Omicienski , Andre dos Santos , Ling Liu , Calton Pu

Proceedings of the 2002 workshop on New security paradigms September 2002

As applications enabled by the Internet become information rich, ensuring access to quality information in the presence of potentially malicious entities will be a major challenge. Denial of information (DoI) attacks attempt to degrade the quality of information by deliberately introducing noise that appears to be useful information. The mere availability of information is insufficient if the user must find a needle in a haystack of noise that is created by an adversary to hide critical informat ...

34 Intrusion detection and response: An empirical analysis of NATE: Network Analysis of Anomalous Traffic Events 77%

Carol Taylor , Jim Alves-Foss

Proceedings of the 2002 workshop on New security paradigms September 2002

This paper presents results of an empirical analysis of NATE (Network Analysis of Anomalous Traffic Events), a lightweight, anomaly based intrusion detection tool. Previous work was based on the simulated Lincoln Labs data set. Here, we show that NATE can operate under the constraints of real data inconsistencies. In addition, new TCP sampling and distance methods are presented. Differences between real and simulated data are discussed in the course of the analysis.

35 Computation in a distributed information market 77%

Joan Feigenbaum , Lance Fortnow , David M. Pennock , Rahul Sami

Proceedings of the 4th ACM conference on Electronic commerce June 2003

According to economic theory supported by empirical and laboratory evidence, the equilibrium price of a financial security reflects all of the information regarding the security's value. We investigate the computational process on the path toward equilibrium, where information distributed among traders is revealed step-by-step over time and incorporated into the market price. We develop a simplified model of an information market, along with trading strategies, in order to formalize the computat ...

36 Technical papers: consistency management and quality assurance: Automated support for classifying software failure reports 77%

Andy Podgurski , David Leon , Patrick Francis , Wes Masri , Melinda Minch , Jiayang Sun , Bin Wang

This paper proposes automated support for classifying reported software failures in order to facilitate prioritizing them and diagnosing their causes. A classification strategy is presented that involves the use of supervised and unsupervised pattern classification and multivariate visualization. These techniques are applied to profiles of failed executions in order to group together failures with the same or similar causes. The resulting classification is then used to assess the frequency and s ...

37 Poster papers: Incremental context mining for adaptive document classification 77%

Rey-Long Liu , Yun-Ling Lu

Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining July 2002

Automatic document classification (DC) is essential for the management of information and knowledge. This paper explores two practical issues in DC: (1) each document has its *context* of

discussion, and (2) both the content and vocabulary of the document database is intrinsically *evolving*. The issues call for *adaptive document classification* (ADC) that adapts a DC system to the evolving contextual requirement of each document category, so that input documents may be classifie ...

38 Poster papers: CVS: a Correlation-Verification based Smoothing technique on information retrieval and term clustering 77%

Christina Yip Chung , Bin Chen

Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining July 2002

As information volume in enterprise systems and in the Web grows rapidly, how to accurately retrieve information is an important research area. Several corpus based smoothing techniques have been proposed to address the data sparsity and synonym problems faced by information retrieval systems. Such smoothing techniques are often unable to discover and utilize the correlations among terms.We propose CVS, a Correlation-Verification based Smoothing method, that considers co-occurrence information i ...

39 Poster papers: Topics in 0--1 data 77%

Ella Bingham , Heikki Mannila , Jouni K. Seppänen

Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining July 2002

Large 0--1 datasets arise in various applications, such as market basket analysis and information retrieval. We concentrate on the study of topic models, aiming at results which indicate why certain methods succeed or fail. We describe simple algorithms for finding topic models from 0--1 data. We give theoretical results showing that the algorithms can discover the epsilon-separable topic models of Papadimitriou et al. We present empirical results showing that the algorithms find natural topics ...

40 Poster papers: Collaborative crawling: mining user experiences for topical resource discovery 77%

Charu C. Aggarwal

Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining July 2002

The rapid growth of the world wide web had made the problem of topic specific resource discovery an important one in recent years. In this problem, it is desired to find web pages which satisfy a predicate specified by the user. Such a predicate could be a keyword query, a topical query, or some arbitrary contraint. Several techniques such as focussed crawling and intelligent crawling have recently been proposed for topic specific resource discovery. All these crawlers are *linkage based*, ...

Search Results

Search Results for: **[data mining <and> conditional probability]**
Found **89** of **121,059 searched.**

Search within Results

> Advanced Search

> Search Help/Tips

Sort by: Title Publication Publication Date Score Binder

Results 1 - 20 of 89 short listing

Prev Page 1 2 3 4 5 Next Page

1 Learning and making decisions when costs and probabilities are both unknown 89%
Bianca Zadrozny , Charles Elkan
Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining August 2001
In many data mining domains, misclassification costs are different for different examples, in the same way that class membership probabilities are example-dependent. In these domains, both costs and probabilities are unknown for test examples, so both cost estimators and probability estimators must be learned. After discussing how to make optimal decisions given cost and probability estimates, we present decision tree and naive Bayesian learning methods for obtaining well-calibrated probability ...

2 Extending naïve Bayes classifiers using long itemsets 89%
Dimitris Meretakis , Beat Wüthrich
Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining August 1999

3 A Bayesian decision model for cost optimal record matching 85%
V. S. Verykios , G. V. Moustakides , M. G. Elfeky
The VLDB Journal — The International Journal on Very Large Data Bases May 2003
Volume 12 Issue 1
In an error-free system with perfectly clean data, the construction of a global view of the data consists of linking - in relational terms, joining - two or more tables on their key fields. Unfortunately, most of the time, these data are neither carefully controlled for quality nor necessarily defined commonly across different data sources. As a result, the creation of such a global data view resorts to approximate joins. In this paper, an optimal solution is proposed for

the matching or the lin ...

4 Research sessions: data mining: Mining long sequential patterns in a noisy environment 85%
Jiong Yang , Wei Wang , Philip S. Yu , Jiawei Han
Proceedings of the 2002 ACM SIGMOD international conference on Management of data June 2002
Pattern discovery in long sequences is of great importance in many applications including computational biology study, consumer behavior analysis, system performance analysis, etc. In a noisy environment, an observed sequence may not accurately reflect the underlying behavior. For example, in a protein sequence, the amino acid N is likely to mutate to D with little impact to the biological function of the protein. It would be desirable if the occurrence of D in the observation can be related to ...

5 Cross-sell: a fast promotion-tunable customer-item recommendation method based on conditionally independent probabilities 84%
Brendan Kitts , David Freed , Martin Vrieze
Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining August 2000

6 Data mining of multidimensional remotely sensed images 84%
Robert F. Cromp , William J. Campbell
Proceedings of the second international conference on Information and knowledge management December 1993

7 Filtering and retrieval models: Collaborative filtering via gaussian probabilistic latent semantic analysis 82%
Thomas Hofmann
Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval July 2003
Collaborative filtering aims at learning predictive models of user preferences, interests or behavior from community data, i.e. a database of available user preferences. In this paper, we describe a new model-based algorithm designed for this task, which is based on a generalization of probabilistic latent semantic analysis to continuous-valued response variables. More specifically, we assume that the observed user ratings can be modeled as a mixture of user communities or interest groups, where ...

8 Classification: Using conjunction of attribute values for classification 82%
Mukund Deshpande , George Karypis
Proceedings of the eleventh international conference on Information and knowledge management November 2002
Advances in the efficient discovery of frequent itemsets have led to the development of a number of schemes that use frequent itemsets to aid developing accurate and efficient classifiers. These approaches use the frequent itemsets to generate a set of *composite features* that expand the dimensionality of the underlying dataset. In this paper, we build upon this work and (i) present a variety of schemes for composite feature selection that achieve a substantial reduction in the number of f ...

9 Search 1: Probabilistic query expansion using query logs 82%
Hang Cui , Ji-Rong Wen , Jian-Yun Nie , Wei-Ying Ma
Proceedings of the eleventh international conference on World Wide Web May 2002
Query expansion has long been suggested as an effective way to resolve the short query and word mismatching problems. A number of query expansion methods have been proposed in traditional information retrieval. However, these previous methods do not take into account the specific characteristics of web searching; in particular, of the availability of large amount of user interaction information recorded in the web query logs. In this study, we propose a new method for query expansion based on qu ...

10 Computing curricula 2001 82%
Journal on Educational Resources in Computing (JERIC) September 2001

11 Hierarchy-based mining of association rules in data warehouses 82%
Giuseppe Psaila , Pier Luca Lanzi
Proceedings of the 2000 ACM symposium on Applied computing March 2000

12 IR theory: Table extraction using conditional random fields 80%
David Pinto , Andrew McCallum , Xing Wei , W. Bruce Croft
Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval July 2003
The ability to find tables and extract information from them is a necessary component of data mining, question answering, and other information retrieval tasks. Documents often contain tables in order to communicate densely packed, multi-dimensional information. Tables do this by employing layout patterns to efficiently indicate fields and records in two-dimensional form.Their rich combination of formatting and content present difficulties for traditional language modeling techniques, however. T ...

13 Web page classification: Web site mining: a new way to spot competitors, customers and suppliers in the world wide web 80%
Martin Ester , Hans-Peter Kriegel , Matthias Schubert
Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining July 2002
When automatically extracting information from the world wide web, most established methods focus on spotting single HTML-documents. However, the problem of spotting complete web sites is not handled adequately yet, in spite of its importance for various applications. Therefore, this paper discusses the classification of complete web sites. First, we point out the main differences to page classification by discussing a very intuitive approach and its weaknesses. This approach treats a web site a ...

14 The true lift model: a novel data mining approach to response modeling in database marketing 80%
Victor S. Y. Lo
ACM SIGKDD Explorations Newsletter December 2002
Volume 4 Issue 2
In database marketing, data mining has been used extensively to find the optimal customer targets so as to maximize return on investment. In particular, using marketing campaign data, models are typically developed to identify characteristics of customers who are most likely to

respond. While these models are helpful in identifying the likely responders, they may be targeting customers who have decided to take the desirable action or not regardless of whether they receive the campaign contact (e ...

15 Searching for dependencies at multiple abstraction levels 80%
Toon Calders , Raymond T. Ng , Jef Wijsen
ACM Transactions on Database Systems (TODS) September 2002
Volume 27 Issue 3
The notion of roll-up dependency (RUD) extends functional dependencies with generalization hierarchies. RUDs can be applied in OLAP and database design. The problem of discovering RUDs in large databases is at the center of this paper. An algorithm is provided that relies on a number of theoretical results. The algorithm has been implemented; results on two real-life datasets are given. The extension of functional dependency (FD) with roll-ups turns out to capture meaningful rules that are outsi ...

16 Cost/benefit based adaptive dialog: case study using empirical medical practice norms and intelligent split menus 80%
Jim Warren
Australian Computer Science Communications , Proceedings of the 2nd Australasian conference on User interface January 2001
Volume 23 Issue 5
The notion of an adaptive user interface, one that accommodates user needs based on knowledge of the task at hand, is compelling but difficult to make practical. This paper examines models of the utility (as balancing of cost and benefit) in the initiation of task-specific dialog based on conditional probability of user goals in context. Illustrations in this paper are based on an empirical model of General Practice (GP) medicine as derived from a large database of GP/patient encounters. Applica ...

17 Evolving data mining into solutions for insights: Data-driven evolution of data mining algorithms 80%
Padhraic Smyth , Daryl Pregibon , Christos Faloutsos
Communications of the ACM August 2002
Volume 45 Issue 8
Fundamentally, these algorithms are driven by the nature of the data being analyzed, in both scientific and commercial applications.

18 Reports from KDD-2001: KDD Cup 2001 report 80%
Jie Cheng , Christos Hatzis , Hisashi Hayashi , Mark-A. Krogel , Shinichi Morishita , David Page , Jun Sese
ACM SIGKDD Explorations Newsletter January 2002
Volume 3 Issue 2
This paper presents results and lessons from KDD Cup 2001. KDD Cup 2001 focused on mining biological databases. It involved three cutting-edge tasks related to drug design and genomics.

19 Theory of keyblock-based image retrieval 80%
ACM Transactions on Information Systems (TOIS) April 2002
Volume 20 Issue 2

The success of text-based retrieval motivates us to investigate analogous techniques which can support the querying and browsing of image data. However, images differ significantly from text both syntactically and semantically in their mode of representing and expressing information. Thus, the generalization of information retrieval from the text domain to the image domain is non-trivial. This paper presents a framework for information retrieval in the image domain which supports content-based q ...

20 Mining web logs for prediction models in WWW caching and prefetching 80%
Qiang Yang , Haining Henry Zhang , Tianyi Li
Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining August 2001
Web caching and prefetching are well known strategies for improving the performance of Internet systems. When combined with web log mining, these strategies can decide to cache and prefetch web documents with higher accuracy. In this paper, we present an application of web log mining to obtain web-document access patterns and use these patterns to extend the well-known GDSF caching policies and prefetching policies. Using real web logs, we show that this application of data mining can achieve dr ...

Results 1 - 20 of 89 short listing
Prev Page 1 2 3 4 5 Next Page

WEST

Generate Collection Print

L16: Entry 6 of 9 File: USPT May 8, 2001

DOCUMENT-IDENTIFIER: US 6230153 B1
TITLE: Association rule ranker for web site emulation

Brief Summary Text (3):
The present invention relates to applying data mining association rules to sessionized web server log data. More particularly, the invention enhances data mining rule discovery as applied to log data by reducing large numbers of candidate rules to smaller rule sets.

Brief Summary Text (5):
Traditionally, discovery of association rules for data mining applications has focused extensively on large databases comprising customer data. For example, association rules have been applied to databases consisting of "basket data"--items purchased by consumers and recorded using a bar-code reader--so that the purchasing habits of consumers can be discovered. This type of database analysis allows a retailer to know with some certainty whether a consumer who purchases a first set of items, or "itemset," can be expected to purchase a second itemset at the same time. This information can then be used to create more effective store displays, inventory controls, or marketing advertisements. However, these data mining techniques rely on randomness, that is, that a consumer is not restricted or directed in making a purchasing decision.

Brief Summary Text (6):
When applied to traditional data such as conventional consumer tendencies, the association rules used can be order-ranked by their strength and significance to identify interesting rules (i.e. relationships.) But this type of sorting metrics is less applicable to sessionized web site data because site imposed associations exist within the data. Imposed associations may be constraints uniformly imposed on visitors to the web site. For example, to determine a relationship between site pages that web site visitors (visitors) find "interesting" using traditional data mining association rules, a researcher might look at pages that have strong link associations. However, for typical web site data, this type of association rule would probably be meaningless because of the site's inherent topology as discussed below.

Brief Summary Text (8):
For example, association rules can be used to identify unsafe patterns of sessionized visits to a web site. Such rules deliver statements of the form "75% of visits from referrer A belong to segment B." Traffic flow patterns can also be uncovered in the form of statements such as "45% of visits to page A also visit page B." However, such rules that characterize behavior due to intentionality of the visitor will tend to be overwhelmed by rules that are due to the traffic flow patterns imposed upon the visitor by the site topology. Therefore, sorting these rules in the conventional manner will place high importance on rules of the form "100% of visitors that invoked URL A also visited URL B." When a visitor's conduct is dominated by the web site topology, rules emanating from such conduct need to be discounted.

Brief Summary Text (9):
Thresholding out the strongest associations between web site pages is neither practical nor desirable, and manually wading through mined association rules for such associations would be excruciatingly tedious and defeat the basic premise upon which data mining was developed.

Brief Summary Text (14):
In another embodiment, the invention may be implemented to provide a method to sort association rules by their relative empirical frequency (relevance), or support, within a database comprising URL data. This relevance ranking is dependant upon the URLs

constituting a complete set of events, and ranks rules where the relevance of each data set is measured by comparing its associational support against the reference given by an emulated distribution. In another embodiment, rules within a set of rules may be compared. The degree deviation of the relevance, or likelihood. of a rule is compared to a reference, such as the number 1, to determine peaks and lows. These peaks and lows are used to determine whether the behavior of actual users compares favorably with the behavior of emulated users. In another embodiment, these rules may be further sorted to determine point-by-point relevance information to distinguish rules that share a common likelihood ratio yet have different supports.

Brief Summary Text (15):
In another embodiment. associations may be ranked even if the URLs comprise an incomplete system of events that may render an emulated choice non-mutually exclusive. In this case, the events are converted into a probability distribution and sorted. In still another embodiment, the converted events may be sorted using more sensitive associations to seek out rules that have unusual levels of support compared to a baseline reference distribution. In another embodiment, association rules may be ranked by their confidence to estimate these conditional probabilities.

Brief Summary Text (18):
The invention affords its users with a number of distinct advantages. One advantage is that the invention provides a way to avoid the necessity of storing massive amounts of historical URL data used to make future comparisons regarding the actions of a user traversing a web site. Another advantage is that the invention reduces the computational time required to process URL data and associations.

Detailed Description Text (13):
Association rules find regularities between sets of items, for example, when an association rule A.fwdarw.B indicates that transactions of a database which contain A also contain B. Either the left hand side ("antecedent" or "head") or the right hand side ("consequent" or "body") can comprise multiple events. Rules of the form u.sub.1 u.sub.2.fwdarw.u.sub.3 u.sub.4 u.sub.5 may be encountered. A rule A.fwdarw.B is defined as having a confidence c% over a set of sessions if c% of the sessions that contain A also contain B, and support s if s% of all sessions contain both A and B.

Detailed Description Text (14):
Efficient algorithms for finding association rules have been provided for mining large databases such as discussed in Agrawal et al., "Fast Discovery of Association Rules," Advances in Knowledge Discovery and Data Mining, Fayyad, U. M. et al. eds., AAAI Press/The MIT Press, Menlo Park, Calif., 1996. However, when applied to web server data, the problem arises that an abundant set of rules must be "distilled" to a manageable size. One way is to rank order rules according to measures of "relevance," "strength," or "importance." One measure of relevance is the support s. A useful measure of strength is given by the confidence c. Other candidate measures are the product of the two, such as cs, as well as c log c and s log s. In conventional transactional databases, these measures can be meaningful, as s measures the portion of transactions in which a rule is relevant, and c gives a direct measure of the associational strength.

Detailed Description Text (15):
However, when used to rank order rules over URLs gleaned from sessionized web server log data, ranking by confidence and support can yield poor results. This is the case when association rules are used to analyze traffic flow patterns of visits to a site, and then those traffic flow patterns are used to infer regularities about the preferences and intentionality of the visitors. Association rules based on confidence and support detect regularities in traffic flow regardless of whether they are due to intentionality on the part of the visitor, or due to forced paths imposed upon the visitor by the web site structure. A rule with substantial support s and strong confidence c can be uninteresting. This follows from what we know about the web site construction, because essentially all visits are subject to certain traffic flow constraints, that may provide little option for choice.

Detailed Description Text (21):
The Web Walker Emulator incorporated by reference above may be used to implement the methods of the present invention. In one embodiment, the Web Walker Emulator is a method for creating a probabilistic generative model of a web site that simulates the behavior of visitors traversing through the site. This simulation "emulates" the behavior of actual visitors to a web site. The parameterization of the simulation can be adjusted in one embodiment such that these "emulated" visitors display behavior that

is substantially indistinguishable from those of actual users (or a subset thereof) with respect to population statistics observed over their respective traffic patterns. Or, in another embodiment, it can be tuned to display hypothetical behavior such as visitors acting without evidence of intentional choice. Tracking the site usage traffic of emulated visitors may yield a set of reference distributions ("emulated distributions") against which may be compared the site usage distributions obtained for actual users. The emulated distributions are used to implement estimation methods which measure relative information content. The Kullback-Liebler Information Criterion and the Bayesian criteria, widely known to those schooled in the art, are two such estimation methods. The result is a set of reference distributions against which the distributions obtained for actual users may be compared.

Detailed Description Text (24):
In the present invention, a reference distribution allows powerful and general-purpose information theoretic statistics to be applied as discussed below for extracting information from a distribution of interest. The Kullback-Liebler Information Criterion (KLIC) mentioned above is one such method that can be used by the present invention for discriminating between distributions. In particular, it measures the directional divergence between two distributions, meaning that the measure is not symmetric. Although it is not a distance measure, it is sometimes referred to as the "KL-distance." It is also easy to construct a variation of the KLIC that yields a non-directional pseudo-distance measure (cf.[Ullah, A., "Entropy, Divergence and Distance Measures with Econometric Applications," Working Paper in Economics, Department of Economics, University of California, Riverside, Riverside, Calif., Journal of Statistical Planning and Inference, 49:137-162, 1996]). For background on the KLIC see White, H., "Parametric Statistical Estimation with Artificial Neural Networks: A Condensed Discussion," From Statistics to Neural Networks: Theory and Pattern Recognition Application, V. Cherkassky, J. H. Friedman and H. Wechsler eds., 1994 and White, H., "Parametric Statistical Estimation with Artificial Neural Networks," P. Smolensky, M. C. Mozer and D. E. Rumelhart eds., Mathematical Perspectives on Neural Networks, L. Erlbaum Associates (to appear), Hilldale, N.J., 1995. For an elegant and concise overview of distributional information measures in general, see Ullah, A., 1996, supra. A brief introduction of KLIC is provided below.

Detailed Description Text (37):
By comparison, the KLIC is a relative measure of information available for distinguishing a target distribution from a reference distribution. It's absolute value is minimized when the target is indistinguishable from the reference--in this case knowing the reference implies knowing the target. For a finite system in which each event is equally likely under the reference distribution, the KLIC is equal to minus the Shannon-Wiener entropy (discussed above) of the target distribution plus a constant. In the preferred embodiment, the present invention requires KLIC (relative entropy) because traditional entropy measure methods rank superfluous association rules highly, exactly the problem that the present invention addresses. One reason superfluous association rules may be highly ranked is because even "randomized" visitor behavior can be highly structured--therefore, have low entropy--due to traffic flow constraints imposed by the web sit topology.

Detailed Description Text (38):
For additional background on the KLIC, White, H., 1994, supra and White, H., 1995, supra, may be consulted and for a concise yet comprehensive survey of KLIC compared and contrasted with other information measures (including Shannon-Wiener information and mutual information) see Ullah, A., 1996, supra.

Detailed Description Text (48):
Consider the task of comparing user behavior from historical data with current day behavior. As a simple means of accomplishing this comparison, historical data can be saved and used for future comparisons. However, this approach has several drawbacks:

Detailed Description Text (52):
Having "emulated users" with the same behavioral characteristics as historical users allows us to evaluate an arbitrary set of statistics at a later time, including statistics that were invented after the historical data was observed. It is possible to create hypothetical situations that were not presented to the historical users, and computationally "imagine" what behaviors historical users night have exhibited if subjected to the hypothetical set of choices. In the present invention. "emulation" combined with "simulation" allows hypothetical situations to be considered, such as, "how would last year's users react to this year's web site structure?" This behavior can then be used as a reference distribution for comparison against this year's

behavior. The following discussion discloses the methods of the present invention that may be used by the Web Walk Emulator for detecting meaningful URL-URL associations.

Detailed Description Text (58):
A rule such A.fwdarw.B.sub.1 can be evaluated over different realizations of the same type of data, such as that produced by different realizations (e.g., as provided by a generative model such as the Web Walk Emulator, or, as observed from the same web site over a different time span) and stored in the database of step 204. If R(A,B.sub.1) gives the support of this rule as measured over emulated visits (henceforth we refer to it as "emulated support") in step 206, two probability distributions over n events are considered that can be compared via relative entropy, namely, P.sub.AB =P(A,B.sub.1),P(A,B.sub.2), . . . P(A,B.sub.n) and R.sub.AB =R(A,B.sub.1),R(A,B.sub.2), . . . , R(A,B.sub.n). Further, K(P.sub.AB :R.sub.AB)=0, where it is some constant value, if and only if these distributions are identical. One way to apply this is to compare P.sub.AB with a different set of association rules, say P.sub.AC =P(A,C.sub.1),P(A,C.sub.n), . . . ,P(A,C.sub.n,), for some integer m and complete system of events C=(C.sub.1, C.sub.2, . . . , C.sub.n,), by computing K(P.sub.AC :R.sub.AC) and comparing with K(P.sub.AB :R.sub.AB) in step 208. If K(P.sub.AC :R.sub.AC)>K(P.sub.AB :R.sub.AB), then association rules applied to the system of events C have higher relevance on average (as compared against the backdrop of the reference R.sub.AC) than that observed for rules over B (as compared against the backdrop of the reference R.sub.AB). The method ends in step 210.

Detailed Description Text (60):
The ranking method discussed above with respect to FIG. 2 compares the relevance of two sets of rules in which the consequents of the rules comprise a complete set of events, where the relevance of each set is measured by comparing its associational support against the reference given by an emulated distribution. However, rules within the same set may also be compared as shown in FIG. 3. Relative entropy is a measure of "expected" information content for discriminating between two distributions--i.e., it is an average value of a pointwise measure. This pointwise measure can be used to compare individual rules within a set of rules. More, precisely: it can be used to compare measures over a set of rules, given that these measures comprise a probability distribution.

Detailed Description Text (71):
This compares the pairs of rules {(A.fwdarw.D.sub.1,A.fwdarw.{character pullout}D.sub.1),(A.fwdarw.D.sub.2,A.fwdarw.{character pullout}D.sub.2) . . . {(A.fwdarw.D.sub.m,A.fwdarw.{character pullout}D.sub.m),} with each other on the basis of whether their support over one data set is unusually high (respectively, low) as compared with the support as evaluated over a data set representing a baseline reference distribution. The method ends in step 514.

Detailed Description Text (72):
Both of the sorting method notations expressed in this section and FIG. 5A are applications of general methods of converting each rule into a corresponding distribution, and then using a distributional measure (average likelihood ratio in (A), and relative entropy in (B)) to compare the resulting distribution with a baseline reference. However, the following quantities may be sorted instead as shown in steps 610 and 612 of FIG. 6:

Detailed Description Text (74):
Statement (C) may be interpreted as seeking out rules that have unusual levels of support compared to a baseline reference distribution, regardless of whether or not the rules are highly supported in the available data as determined by P. Statement (D) also seeks out rules having unusually high or low support, but weights them according to their support over the observed data as determined by P such that given two rules with identical likelihood ratios, the one with greater support will be sorted closer to the head (or tail) of the rank ordering. The method ends in step 614.

Detailed Description Text (76):
In another embodiment. and under the appropriate conditions (e.g., sufficient data, stationary data generating process), association rules' measures of "confidence" can be used in one method to estimate conditional probabilities. In particular, the confidence of rule A.fwdarw.B gives a useable estimate of the conditional probability P(A.vertline.B). The same techniques as described immediately above may be applied for rule support to compare the confidence of rules against a baseline reference distribution. Relationships such as defined in statements (C) and (D) above are easily applied to evaluating rule confidence. With substitution of the appropriate conditional

probabilities, the relationships and the rules are sorted in steps 706 and 708 where:

Detailed Description Text (78):
Sorting likelihood ratios as described above are equivalent to traversing the distribution P.sub.AB /R.sub.AB and looking for places where it deviates significantly from 1. Peaks (ratios much greater than 1) show where the confidence of rules under P.sub.AB is significantly greater than what is suggested by R.sub.AB, and dips (ratios close to 0) show where the confidence under P.sub.AB is unusually lower than what is suggested by R.sub.AB. Comparatively speaking, the interpretation of the relationship statement (E) is not as tidy because the conditional probabilities do not in general lend themselves to forming a probability distribution; for each i in statement (E) simply delivers a pointwise measure of the information content for discriminating between the two distributions {P(A.vertline.D.sub.i),P(A.vertline.{character pullout}D.sub.i)} and {R(A.vertline.D.sub.i),R(A.vertline.{character pullout}D.sub.i)}. The relationships in statement (F) add the benefit of giving emphasis to rules with greater support, which is ideally suited to the determining applications for which these techniques are intended. The method ends in step 710.

Current US Original Classification (1):
707/2

Current US Cross Reference Classification (2):
707/200

Current US Cross Reference Classification (3):
707/7

Other Reference Publication (6):
J.S. Park, et al, "Efficient Parallel Data Mining For Association Rules", IBM Research Report, RJ 20156, Aug. 1995.

Other Reference Publication (10):
Agrawal et al, "Mining Association Rules Between Sets of Items In Large Databases", Proc. 1993 ACM SIGMOD Conf. pp. 207-216, 1993.

CLAIMS:

1. A method for sorting data mining association rules, the method comprising:

identifying statistically significant relationships within a cumulated distribution of data, the significant relationships represented by association rules; and

separating meaningful association rules from unmeaningful association rules using an emulated distribution of the data as a reference, wherein said emulated distribution is based upon emulated events that are different than actual events.

16. The method recited in claim 2, wherein association rules having high levels of support compared to the emulated distribution are ranked highest, regardless of whether the association rules are highly supported in the uniform resource locator data as determined by P, and where P is a probability of an occurrence of the association rule.

18. A method for sorting data mining association rules, the method comprising:

identifying statistically significant relationships within a cumulated distribution of data, the significant relationships represented by association rules; and

separating meaningful association rules from unmeaningful association rules using an emulated distribution of the data as a reference by sorting the rules by their support within the distribution of data,

wherein the uniform resource locator data does not comprise a system of events and is sorted by m sets of uniform resource locator data, where {P(A,D.sub.i)/R(A,D.sub.i)}, i=], 2, . . . , m, and D.sub.i corresponds to sets of uniform resource locator data 1 to m.

20. A method for sorting data mining association rules, the method comprising:

identifying statistically significant relationships within a cumulated distribution of

data, the significant relationships represented by association rules; and

separating meaningful association rules from unmeaningful association rules using an emulated distribution of the data as a reference by sorting the rules by their confidence within the distribution of data,

sorting of the association rules comprising ranking the rules by their confidence, where {P(A.vertline.D.sub.i)/R(A.vertline.D.sub.i)}, i=1, 2, . . . , m, where P is a probability of an occurrence of an association rule, and where two association rules with identical P values are further sorted so that the rule with greater support in the emulated data is sorted higher than the rule with the lesser support.

22. An article of manufacture comprising a data storage medium tangibly embodying a program of machine-readable instructions executable by a digital processing apparatus to perform a method for sorting data mining association rules, the method comprising:

identifying statistically significant relationships within a cumulated distribution of uniform resource locator data, the significant relationships represented by association rules; and

separating meaningful association rules from unmeaningful association rules using an emulated distribution of the uniform resource locator data as a reference, wherein said emulated distribution is based upon emulated events that are different than actual events.

37. The article recited in claim 22, wherein association rules having high levels of support compared to the emulated distribution are ranked highest, regardless of whether the association rules are highly supported in the uniform resource locator data as determined by P, and where P is a probability of an occurrence of the association rule.

39. An article of manufacture comprising a data storage medium tangibly embodying a program of machine-readable instructions executable by a digital processing apparatus to perform a method for sorting data mining association rules, the method comprising:

identifying statistically significant relationships within a cumulated distribution of uniform resource locator data, the significant relationships represented by association rules; and

separating meaningful association rules from unmeaningful association rules using an emulated distribution of the uniform resource locator data as a reference, wherein said emulated distribution is based upon emulated events that are different than actual events;

wherein the uniform resource located data does not comprise a system of events and is sorted by m sets of uniform resource locator data, where {P(A,D.sub.i)/R(A,D.sub.i)}, i=1, 2, . . . , m, and D.sub.i corresponds to sets of uniform resource locator data 1 to m.

41. An article of manufacture comprising a data storage medium tangibly embodying a program of machine-readable instructions executable by a digital processing apparatus to perform a method for sorting data mining association rules, the method comprising:

identifying statistically significant relationships within a cumulated distribution of uniform resource locator data, the significant relationships represented by association rules; and

separating meaningful association rules from unmeaningful association rules using an emulated distribution of the uniform resource locator data as a reference, wherein said emulated distribution is based upon emulated events that are different than actual events;

sorting of the association rules comprising ranking the rules by their confidence, where {P(A.backslash.D.sub.i)/R(A.backslash.D.sub.i)}, i=1, 2, . . . , m, where P is a probability of an occurrence of an association rule, and where two association rules with identical P values are further sorted so that the rule with greater support in the emulated data is sorted higher than the rule with the lesser support.

43. An apparatus to sort data mining association rules, comprising:

a processor;

a database including URL data;

circuitry to communicatively couple the processor to the database;

storage communicatively accessible by the processor; the processor sorting mining association rules by:

identifying statistically significant relationships within a cumulated distribution of uniform resource locator data, the significant relationships represented by association rules; and

separating meaningful association rules from unmeaningful association rules using an emulated distribution of the uniform resource locator data as a reference, wherein said emulated distribution is based upon emulated events that are different than actual events.

58. The apparatus recited in claim 43, wherein association rules having high levels of support compared to the emulated baseline reference distribution are ranked highest, regardless of whether the association rules are highly supported in the uniform resource locator data as determined by P, and where P is a probability of an occurrence of the association rule.

60. An apparatus to sort data mining association rules, comprising:

a processor;

a database including URL data;

circuitry to communicatively couple the processor to the database;

storage communicatively accessible by the processor; the processor sorting mining association rules by:

identifying statistically significant relationships within a cumulated distribution of uniform resource locator data, the significant relationships represented by association rules; and

separating meaningful association rules from unmeaningful association rules using an emulated distribution of the uniform resource locator data as a reference, wherein said emulated distribution is based upon emulated events that are different than actual events;

where the uniform resource locator data does not comprise a system of events and is sorted by m sets of uniform resource locator data, where {P(A,D.sub.i)/R(A,D.sub.i)}, i=1, 2, . . . , m, and D.sub.i corresponds to sets of uniform resource data 1 to m.

62. An apparatus to sort data mining association rules, comprising:

a processor;

a database including URL data;

circuitry to communicatively couple the processor to the database;

storage communicatively accessible by the processor; the processor sorting mining association rules by:

identifying statistically significant relationships within a cumulated distribution of uniform resource locator data, the significant relationships represented by association rules; and

separating meaningful association rules from unmeaningful association rules using an emulated distribution of the uniform resource locator data as a reference, wherein said emulated distribution is based upon emulated events that are different than actual events;

sorting of the association rules comprising ranking the rules by their confidence, where {P(A.backslash.D.sub.i)/R(A.backslash.D.sub.i)}, i=1,2, . . . , m, where P is a probability of an occurrence of an association rule, and where two association rules with identical P values are further sorted higher than the rule with the lesser support.

64. An apparatus for sorting data mining association rules, comprising:

means for storing URL data;

means for processing the URL data by:

identifying statistically significant relationships within a cumulated distribution of uniform resource locator data, the significant relationships represented by association rules; and

separating meaningful association rules from unmeaningful association rules using an emulated distribution of the uniform resource locator data as a reference, wherein said emulated distribution is based upon emulated events that are different than actual events.

WEST

Freeform Search

Database: US Patents Full-Text Database
US Pre-Grant Publication Full-Text Database
JPO Abstracts Database
EPO Abstracts Database
Derwent World Patents Index
IBM Technical Disclosure Bulletins

Term: L15 and compar$

Display: 50 **Documents in Display Format:** FRO **Starting with Number** 1

Generate: ○ Hit List ◉ Hit Count ○ Side by Side ○ Image

Search | Clear | Help | Logout | Interrupt

Main Menu | Show S Numbers | Edit S Numbers | Preferences | Cases

Search History

DATE: Monday, September 08, 2003 Printable Copy Create Case

Set Name side by side	Query	Hit Count	Set Name result set
DB=USPT; PLUR=YES; OP=OR			
L16	L15 and compar$	9	L16
L15	L14 and (conditional near probability)	9	L15
L14	L13 and pattern$	71	L14
L13	L12 and (set near item$)	84	L13
L12	l7 and (data near mining)	388	L12
L11	L10 and (compar$ or match$)	31	L11
L10	L9 and (identifier$ or status)	33	L10
L9	L8 and synchroniz$	40	L9
L8	L7 and (incremental near chang$)	100	L8
L7	((707/$)!.CCLS.)	11256	L7
L6	((717/$)!.CCLS.) and configuration	1681	L6
L5	L3 and (remote near image)	2	L5
L4	L3 (remote near image)	1714	L4
L3	L2 and (707/$.ccls.)	53	L3
L2	medical near image	2041	L2
L1	medical image	570011	L1

END OF SEARCH HISTORY

L16: Entry 5 of 9 File: USPT Aug 21, 2001

DOCUMENT-IDENTIFIER: US 6278997 B1
TITLE: System and method for constraint-based rule mining in large, dense data-sets

Brief Summary Text (5):
Customer purchasing habits can provide invaluable marketing information for a wide variety of applications. This type of data may be known as market basket data. For example, retailers can create more effective store displays and more effectively control inventory than otherwise would be possible if they know that, given a consumer's purchase of a first set of items (a first itemset), the same consumer can be expected, with some degree of likelihood of occurrence, to purchase a particular second set of items (a second itemset) along with the first set of items. In other words, it is helpful from a marketing standpoint to know the association between the first itemset and the second itemset (the association rule) in a given data-set. For example, it would be desirable for a retailer of automotive parts and supplies to be aware of an association rule expressing the fact that 90% of the consumers who purchase automobile batteries and battery cables (the first itemset) also purchase battery post brushes and battery post cleansers (referred to as the "consequent" in the terminology used in the present description). Market basket data is data in which there are one or more data elements representing purchased items, such as bread, milk, eggs, pants, etc., in a transaction, such as an individual consumer purchase. For market basket data, no data element has only a limited predetermined set of values, such as male or female, so that the values occur frequently. For example, the first data element in any transaction may be any item which may be purchased by the consumer so that one can not assume, for example, that the first data element contains a milk item. Thus, since each data element may have a variety of values, the market basket data is not "dense" data.

Brief Summary Text (9):
Not surprisingly, many methods have been developed for mining these large databases. The problem of mining association rules from large databases was first introduced in 1993 at the ACM SIGMOD Conference of Management of Data in a paper entitled, "Mining Association Rules Between Sets of Items in a Large Database" by Rakesh Agrawal, Tomasz Imielinski and Arun Swami. In general, the input, from which association rules are mined, consists of a set of transactions where each transaction contains a set of literals (i.e., items). Thus, let I={1.sub.1, 1.sub.2, . . . 1.sub.m } be a set of literals called items. Let D be a set of transactions, where each transaction T is a set of items such that T.OR right.I. Therefore, a transaction T contains a set A of some items in I if A.OR right.T.

Brief Summary Text (10):
An association rule is an implication of the form A{character pullout}B, where A.OR right.I, B.OR right.I, A.andgate.B=.O slashed. and B is the consequent of the rule. The rule A{character pullout}B holds true in the transaction set D with a confidence "c" if c % of transactions in D that contain A also contain B (i.e., the confidence in the conditional probability p(B.vertline.A)). The rule A{character pullout}B has support "s" in the transaction set D if s transactions in D contain A.orgate.B (i.e., the support is the probability of the intersection of the events). The support s may also be specified as a percentage of the transactions in the data-set that contain A.orgate.B. An example of an association nile is that 30% of the transactions that contain beer and potato chips also contain diapers and that 2% of all transactions contains all of these items. In this example, 30% is the confidence of the association rule and 2% is the support of the rule. The typical problem is to find all of the association rules that satisfy user-specified constraints. As described above, this mining of association rules may be useful, for example, to such applications as market basket analysis, cross-marketing, catalog design, loss-leader analysis, fraud detection, health insurance, medical research and telecommunications diagnosis.

Brief Summary Text (11):
Most conventional data mining systems and methods, such as a method known as Apriori and its descendants, are developed to tackle finding association rules in market basket data which is not dense data. The problem is that these conventional systems, when faced with dense data such as census data, experience an exponential explosion in the computing resources required. In particular, these conventional systems mine all association rules (also referred to simply as rules) satisfying a minimum support constraint, and then enforce other constraints during a post-processing filtering step. Thus, for the dense census data, any transaction containing male or female may be mined. However, this generates too many rules to be useful and takes too much time. During the post-processing, the total number of rules may be reduced by applying a minimum predictive accuracy constraint, such as minimum confidence, lift, interest or conviction. However, even with these additional post-processing constraints, these conventional systems still generate too many rules for dense data which 1) take too long to generate, and 2) can not be easily comprehended by the user of the system.

Brief Summary Text (12):
There are also other conventional data mining systems for "dense" data, such as heuristic or "greedy" rule miners, which try to find any rules which satisfy a given constraint. An example of a greedy miner is a decision tree induction system. These conventional systems generate any rules satisfying the given constraints or a single rule satisfying the constraints, but do not necessarily generate a complete set of rules which may satisfy the given constraints. These conventional systems also do not attempt to determine a "best" rule (e.g., most predictive) so that, at best, an incomplete set of rules, none of which may be a best rule, may be generated which is not useful to the user of the system.

Brief Summary Text (17):
In accordance with the invention, the dense data-set may be processed or mined to generate a set of association rules. First, set enumeration tree is generated level by level. Each node in the set enumeration tree enumerates an association rule which may satisfy the user constraints. Each node in the set enumeration tree is called a group since it implicitly represents the group of association rules that can be enumerated by an sub-node of the node. After each level of the set enumeration tree is generated, rules which satisfy the user constraints are extracted from the rules enumerated by that level. Then, any group which satisfies certain criteria may be pruned from the set enumeration tree. The criteria used to prune a group from the set enumeration tree may include comparing an upper bound on the gap of any rule in the group to the user constraint of minimum gap, comparing an upper bound on the confidence of any rule in the group to the user constraint of minimum confidence and comparing an upper bound on the support of any rule in the group to the user constraint of minimum support as described below in more detail. During the pruning process, either an entire group is pruned or a particular portion of a group known as a tail item may be pruned. It should be noted that the groups within each level of the set enumeration tree are pruned twice, once before and once after the groups are processed to determine the support of the association rules in the groups (also known as group members). To aid the pruning process, a item ordering method may be used which tends to place items and groups which may be prunable underneath the same head item.

Brief Summary Text (20):
In accordance with yet another aspect of the invention, a method for pruning a set enumeration tree used to discover association rules within a dense data is provided in which the set enumeration tree includes one or more groups of items arranged in a tree wherein each item within a group is within a head or a tail of the group. To prune the set enumeration tree, any groups from the set enumeration tree are removed based on a predetermined set of criteria, and then any items in the set enumeration tree are removed from the tail group of each of the one or more groups in the set enumeration tree based on the predetermined set of criteria.

Drawing Description Text (9):
FIG. 8 is a flowchart illustrating a method for re-ordering the tail items in the set enumeration tree in accordance with the invention;

Detailed Description Text (3):
As shown, the operating system of the server computer 14 may include a dense data mining kernel 16 which may be executed by a processor within the server computer 14 as a series of computer-executable instructions. These computer-executable instructions may reside in a memory, for example, in the RAM of the server computer 14. Alternatively, the instructions may be contained on a data storage device with a

computer readable medium, such as a computer diskette 15 shown in FIG. 2. The instructions may also be stored on a DASD array, a magnetic tape, a conventional hard disk drive, electronic read-only memory, an optical storage device, or any other appropriate data storage device. In an illustrative embodiment of the invention, the computer-executable instructions may be lines of compiled C++ language code.

Detailed Description Text (7):
FIG. 1 also illustrates that the client computer 12 may include a mining kernel interface 26 which, like the mining kernel 16, may be implemented in suitable computer program code. Among other things, the interface functions as an input mechanism for establishing certain variables, including a minimum confidence and support value, a minimum gap value, and the other predetermined/user-defined input parameters disclosed below. Further, the client computer 12 preferably includes an output module 28 for outputting /displaying the results stored in the results repository 24 on a graphical display 30, to a printing mechanism 32 or to a data storage medium 34. The functional details of the dense data mining kernel 16 will be described shortly. First, however, to better understand the invention, an example of the benefits of such a system will be described.

Detailed Description Text (13):
For rules to be comparable in the above-described context, they must have equivalent consequents. Therefore, the method in accordance with the invention uses a consequent which is fixed and specified in advance. This fixed consequent setting is quite natural in many applications where the goal is to discover properties of a specific class of interest. This task is sometimes referred to as partial classification and may be applicable in a variety of areas, such as telecommunications service analysis, fraud detection, and targeted marketing. Now, the details and the context of the constraint-based dense data miner in accordance with the invention will be described.

Detailed Description Text (16):
Next, at step 40, the method extracts rules from the remaining groups in the set enumeration tree not previously pruned that are known to have minimum support and minimum confidence. Next, in step 42, the method again prunes group from the data-set to further reduce the total number of association rules which may be mined. Thus, in accordance with the invention, the set enumeration tree may be pruned twice during the data mining process which reduces the total number of association rules generated by the data mining. Once the second pruning has been completed, it is determined whether the set enumeration tree is empty (i.e., there are no more groups in the set enumeration tree to analyze and process) in step 44. If the set enumeration tree is not empty, the method returns to step 34 in which the processing of the data-set continues. If the set enumeration tree is empty, then in step 46, any post-processing of the generated association rules may be completed and the output may be the dense data association rules. Prior to describing each of the individual steps of the method, a description of the constraints used in accordance with the invention will be provided.

Detailed Description Text (17):
The conventional association rule mining problem is to produce all association rules present in a data-set that meet specified minimum support values and then a minimum confidence value may be used to post-process the mined association rules. However, as described above, conventional association rule mining systems experience an exponential explosion the number of association rules returned when the conventional systems are used to mine association rules from dense data. Therefore, an accordance with the invention, additional constraints (i.e., confidence and gap) are used to mine the association rules to render a system for mining association rules for dense data-sets. The constraints used in the system and method for dense data mining in accordance with the invention will now be described, but it should be noted that a variety of other constraints may also be used and therefore the invention should not be limited to the particular constraints described herein.

Detailed Description Text (18):
In accordance with the invention, various constraints may be used to prune the set enumeration tree. First, the mining of rules is restricted to those that have a given consequent c. This restriction is referred to as an item-constraint or a consequent constraint which has been exploited by other conventional systems and methods, but only to reduce the set of frequent itemsets considered prior to the actual data mining. Thus, for these conventional methods, the consequent constraint is used to improve the manner in which the minimum support constraint is exploited. In accordance with the invention, however, the method does not attempt to mine frequent itemsets because frequent itemsets are too numerous in dense data even given this item constraint.

Instead, to reduce the total number of mined association rules, the method in accordance with the invention directly mines rules meeting all of the given constraints. Thus, the consequent constraint is used not only to improve the manner in which minimum support is exploited, but also the manner in which minimum confidence and the minimum gap constraints, as described below, are exploited.

Detailed Description Text (21):
Thus, in accordance with the invention, the method mines all association rules with a given consequent meeting the user-specified minimums on support, confidence, and gap. For the description provided below the following terminology will be followed. The parameter specifying the minimum confidence bound may be referred to as "minconf", the minimum support bound may be referred to as "minsup" and the parameter specifying a minimum gap may be referred to as "mingap". A rule is said to be confident if it has confidence greater than or equal to minconf, and frequent if it has support greater than or equal to minsup. A rule is said to have a large gap when its gap exceeds mingap. Since the consequent is assumed to be fixed, an association rule may be represented as the set of items tested to predict the consequent. Now, an example of dense data-set mining in accordance with the invention will be described.

Detailed Description Text (23):
FIG. 4 is a diagram illustrating an example of a set enumeration tree 50 which may be used to order the data-set to permit the data-set to be mined for association rules. The rule mining problem is then one of searching through the power set of the itemset consisting of all items present in the database for rules which satisfy the minsup, minconf, and mingap constraints. To mine the data-set, Rymon's conventional set-enumeration tree framework as described in an article entitled "Search Through Systematic Set Enumeration" in the Proceedings of the Third International Conference on Principles of Knowledge Representation and Reasoning in 1992, provides a scheme for representing a subset search problem as a tree search problem, allowing pruning rules, as described below, to be defined in a straightforward manner in order to reduce the space of subsets (rules) considered. The idea is to first impose an ordering on the set of items, and then enumerate sets of items according to the ordering as illustrated in FIG. 4.

Detailed Description Text (24):
In FIG. 4, the set enumeration tree 50 is illustrated for a 4-item data set with each item denoted by its position in the ordering. As shown, the 4-item data set may have a null root node 52 and each of the items (1,2,3,4) in the data set may be a child 53 of the head node. In this example, the first item in the set enumeration tree is item 1 so that the children underneath the node containing item 1 are the various combination of items (e.g., 1,2,3 or 1,3,4) which contain item 1. Similarly, underneath the other items are the combinations which contain those items, but which were not contained in the combination under item 1. The set enumeration tree, therefore, lists all of the possible combinations for the items. As is well known, which item is first in the search tree (i.e., at the left hand side) determines which item has a large number of combinations underneath it. This property is exploited by the method in accordance with the invention in order to rapidly prune groups as described below.

Detailed Description Text (25):
For purposes of this method, the terminology and conventional techniques developed in previous works in which one attempted to mine maximal frequent itemsets from large data-sets as a set-enumeration tree search problem may be used. Therefore, each node in the tree may be represented by two itemsets comprising a candidate group, or just group for short. The first itemset, called the head, is simply the itemset (rule) enumerated at the given node. The second itemset, called the tail, is actually an ordered set and consists of those items which can be potentially appended to the head to form any rule appearing as a sub-node. The head and tail of a group g will be denoted as h(g) and t(g), respectively. The order in which tail items appear in t(g) is significant since it reflects how its children are to be expanded as will be described below with reference to FIG. 7. Each child, g.sub.c, of a group, g, is formed by taking an item i.epsilon.t(g) and appending it to h(g) to form h(g.sub.c). Then, t(g.sub.c) is made to contain all items in t(g.sub.p) that follow i in the ordering. Given this child expansion policy, without any pruning of nodes or tail items, the set-enumeration tree enumerates each and every subset exactly once as described above.

Detailed Description Text (27):
The candidate set of a group g may be defined to be the set of itemsets h(g), h(g).orgate.c, h(g).orgate.{i} and h(g).orgate.{i}.orgate.c for all i.epsilon.t(g), h(g).orgate.t(g), and h(g).orgate.t(g).orgate.c. We denote the number of transactions

in the data-set to contain a particular set of items I as sup(I). A group is said to be processed once the method has computed the support of every itemset in its candidate set. The use of well known hash-trees and other implementation details for efficiently computing the support of all itemsets in the candidate sets of several groups may be used in accordance with the invention. Now, the pruning in accordance with the invention will be described.

Detailed Description Text (31):
As shown in FIG. 5, the pruning method 60 is applied for each group g within the data-set G. At step 62, the method determines whether or not the particular group g is prunable. To determining whether a group is prunable, one or more values for the particular group are calculated and compared to the constraints (e.g., minconf, mingap and minsup). The details of determining if a group is prunable will be described below with reference to FIG. 6. If the group g is prunable, then at step 64, the group g is removed from the data-set G and the next group (g+1) is tested to determine whether it is prunable at step 62. If the particular group g is not prunable, then in step 66, the method determines whether or not some of the items in the tail t(g) of group g are prunable using the same method as will be described with reference to FIG. 6. If none of the items in the tail are prunable, then the method loops back to step 62 to test another group. If there are some items in the tail which are prunable, then those prunable items are removed from the tail in step 68 and the method loops back to step 62 to recheck the group g and its tail items. Because fewer tail items can improve the ability of step 62 to determine whether a group can be pruned, whenever a tail item is found to be prunable from a group, the group and all tail items are rechecked. In accordance with the invention, a group may be pruned or, if the group cannot be pruned, some items in its tail may be pruned which significantly reduces the amount of work to locate the rules. Now, a method for determining if a group g is prunable will be described.

Detailed Description Text (32):
FIG. 6 is a flowchart illustrating a method 70 for determining if a group g in the data-set G is prunable in accordance with the invention. To determine the prunability of a group, the method 70 applies pruning rules which compute, for each group g: 1) an upper-bound uconf(g) on the confidence of any rule derivable from g in step 72; 2) an upper-bound ugap(g) on the gap of any derivable rule from g that is frequent in step 74; and 3) an upper-bound usup(g) on the support of any derivable rule in step 76. The method for determining these upper-bounds will be described below. The goal of pruning is to prune a group without affecting the completeness of the search and this goal is accomplished by comparing the calculated values against the user-specified constraints. In particular, in step 78, the method determines if uconf(g) is less than minconf and prunes the group g in step 80 if the condition is true. If uconf(g) is not less than minconf, then the method continues to step 82 in which the method determines if ugap(g) is less than or equal to mingap. If the condition is true, then the group g is pruned in step 80. If ugap(g) is not less than or equal to mingap, then the method, in step 84, determines if usup(g) is less than minsup. If the condition is true, then the group g is pruned in step 80. If usup(g) is not less than minsup, then the method ends. In summary, for each group g, uconf, ugap and usup values are calculated, compared to the corresponding user-specified constraints, and the group is pruned is any one of the above-identified conditions is met.

Detailed Description Text (63):
We lastly discuss how to obtain the value of usup(g), which is an upper-bound on the support of any rule derivable from g. This value is comparatively easy to compute because support is anti-monotone with respect to rule containment. For usup(g), we simply use the value of sup(h(g).orgate.c) if the group is unprocessed, and the tighter value of max(.A-inverted.i.epsilon.t(g), sup(h(g).orgate.{i}.orgate.c)) when the group is processed. Now, a method for determining the next level of the set enumeration tree and a method for set enumeration tree item re-ordering in accordance with the invention will be described.

Detailed Description Text (64):
FIG. 7 is a flowchart illustrating a method 100 for determining the next level of the set enumeration tree in accordance with the invention. The method shown in FIG. 7 is repeated for each candidate group g in the data-set G. At step 102, the tail items for the candidate group are reordered, as described below with reference to FIG. 8, which improves the efficiency of the pruning constraints. Next, in step 104, for each item in the tail of each group in the set enumeration tree, a new candidate group, g', is generated. In particular, the new group, g', may be generated in which h(g')=h(g).orgate.{i} and t(g')={i'} wherein i' comes after i in the ordering. In

accordance with the invention, these new candidate groups may be pruned from the set enumeration tree as described above. If there are no additional items, then the method generates the new groups, g', in step 108 which may then be pruned and processed in accordance with the invention. Next, a method for reordering the tail items in the set enumeration tree in accordance with the invention will be described.

Detailed Description Text (70):
FIG. 9 illustrate a method 130 for preparing the association rules for post-processing in accordance with the invention. Generally, a preferred post-processor may carefully searches the space of sub-rules using another set-enumeration tree search that prunes many rules from consideration. First, many rules without a large gap may be identified simply by comparing them to the others in the mined rule set as shown in step 132. In particular, given the set of mined rules R, the post-processor therefore compares each rule r.sub.1.epsilon.R to every rule r.sub.2 such that r.sub.2.epsilon.R and r.sub.2.OR right.r.sub.1. As set forth in step 134, the post processing method determines if conf(r.sub.1).ltoreq.conf(r.sub.2)+mingap and removes rule r.sub.1 in step 136 if the condition is true because the gap value for r.sub.1 is not sufficiently large (i.e., the rule r.sub.1 does not have a sufficiently higher gap value to warrant keeping the rule). This step requires no database access and it may remove almost all rules that do not have a large gap. In fact, if mingap is set to 0, then this phase removes every such rule. In step 138, the post-processing method checks for more rules to compare and loops back to step 132 if more rules need to be compared. Otherwise the preparation of the association rules for post processing has been completed and the post processing is begun as will now be described with reference to FIG. 10.

Current US Original Classification (1):
707/6

Other Reference Publication (2):
"Mining Association Rules Between Sets of Items In Large Databases", Agrawal et al, Proceedings of the ACM-SIGMOD 1993 Int'l Conference On the Management of Data, Washington, D.C. 1993, pp. 207-216.

Other Reference Publication (5):
"Efficient Parallel Data Mining For Association Rules", Park et al, IBM Research Report, 26 pages, R 20156, Aug. 1995.

Other Reference Publication (12):
"Efficiently Mining Long Patterns From Databses", R. J. Bayardo, Jr., To appear in Proc. of the 1998 ACM-SIGMOD Conference on Management of Data.

Other Reference Publication (16):
"Mining Sequential Patterns", Proc. Of the Int'l Conference on Data Engineering, Taipei, Taiwan, 1995, pp. 3-14.

WEST

Generate Collection Print

L16: Entry 4 of 9 File: USPT Oct 9, 2001

DOCUMENT-IDENTIFIER: US 6301575 B1
TITLE: Using object relational extensions for mining association rules

Abstract Text (1):
A method, apparatus, and article of manufacture for computer-implemented use of object relational extensions for mining association rules. Data mining is performed by a computer to retrieve data from a data store stored on a data storage device coupled to the computer. A multi-column data store organized using a multi-column data model is received. One of the columns in the multi-column data store represents a transaction, and each of the remaining columns in the multi-column data store represents elements of that transaction. A combination operator is performed to obtain candidate itemsets of data from the multi-column data store, each itemset being a combination of a number of rows of the multi-column data store. Large itemsets of data are generated from the candidate itemsets, wherein each itemset has at least a minimum support. Association rules are generated from the large itemsets of data, wherein each association rule has at least a minimum confidence.

Brief Summary Text (3):
This invention relates in general to computer implemented data mining, and in particular to using object relational extensions for mining association rules.

Brief Summary Text (5):
There has been a rapid growth in the automation of data collection procedures in the last decade. This has led to a vast growth in the amount of usable data. Translating this usable data to useful information requires the use of a variety of data mining and knowledge extraction techniques. Accompanying these developments has been the growth of reliable, highly optimized relational database systems. As more and more data stores begin to rely on these database systems, the integration of the mining techniques with the database systems becomes desirable. However, efficient utilization of database systems as mining engines requires some modifications to the relational database system and to data organization.

Brief Summary Text (6):
Data mining is the process of finding interesting patterns in data. Data mining retrieves interesting data from a very large database, such as a database describing existing, past, or potential clients that may have thousands of attributes. A database is a set of records that are described by a set of attributes which have values.

Brief Summary Text (7):
Conventional data mining techniques do not work well on a database with a large number of attributes. In particular, most conventional data mining techniques only work one data in memory. Therefore, if the data is so large that it must be stored other than in memory, the data mining techniques will move data into memory to operate on the data, which is inefficient both in terms of memory usage and time.

Brief Summary Text (8):
The successful automation of data collection and the growth in the importance of information repositories have given rise to numerous data stores, ranging from those of large scientific organizations, banks and insurance companies, to those of small stores and businesses. The abundance of data has required the use of innovative and intricate data warehousing and data mining techniques to summarize and make use of the data.

Brief Summary Text (10):
Some of the new techniques for knowledge extraction are for clustering of data. T. Zhang, R. Ramakrishnan, and M. Livny, Birch, An Efficient Data Clustering Method For Very Large Databases, Proceedings of the 1996 ACM SIGMOD International Conference of

Management of Data, 1996; R. T. Ng and J. Han, Efficient And Effective Clustering Methods For Spatial Data Mining, Proceedings of the 20th International Conference on Very Large Databases, 1994; A. K. Jain and R. C. Dubes, Techniques For Clustering Data, Prentice-Hall, 1988; L. Kaufman and P. J. Rousseeuw, Finding Groups In Data--An Introduction To Cluster Analysis, Wiley, 1990, which are incorporated by reference herein.

Brief Summary Text (11):
Some of the techniques for knowledge extraction are for discovery of association rules, and association rules are derived from and used to represent frequently occurring patterns within the database. R. Agrawal., T. Imielinski, and A. Swami, Mining Association Rules Between Sets Of Items In Large Databases, Proceedings of SIGMOD '93, pages 207-216, May 1993; R. Agrawal and R. Srikant, Fast Techniques For Mining Association Rules, Proceedings of the 20th International Conference on Very Large Databases, September 1994, [hereinafter "Fast Techniques For Mining Association Rules"]; M. Houtsma and A. Swami, Set-Oriented Mining Of Association Rules, Technical Report RJ 9567, IBM Almaden Research Center, October 1993, [hereinafter "Set-Oriented Mining of Association Rules"]; J. S. Park, M. S. Shen, and P. S. Yu, An Effective Hash Based Technique For Mining Association Rules, Proceedings of SIGMOD '95, May 1995; R. Agrawal, H. Mannila, R. Srikant, H. Toivonen, and A. I. Verkamo, Fast Discovery Of Association Rules, Advances in Knowledge Discovery and Data Mining, AAAI/MIT Press, edited by U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, 1995; H. Toivonen, Sampling Large Databases For Association Rules, Proceedings of the 22nd International Conference on Very Large Databases; A. Savasere, E. Omiecinski, and S. Navathe, An Efficient Technique For Mining Association Rules In Large Databases, Proceedings of the 21 st International Conference on Very Large Databases, September 1995; H. Mannila, H. Toivonen, and A. I. Verkamo, Efficient Techniques For Discovering Association Rules, Technical Report WS-94-03, American Association for Artificial Intelligence, 1994; R. Srikant and R. Agrawal, Mining Generalized Association Rules, Proceedings of the 21 st International Conference on Very Large Databases, September 1995; J. Han and Fu, Discovery Of Multiple-Level Association Rules From Large Databases, Proceedings of the 21st International Conference on Very Large Databases, September 1995; J. Han, Y. Cai, and N. Cercone, Data--Driven Discovery Of Quantitative Rules In Relational Databases, IEEE Transactions on Knowledge and Data Engineering, Vol. 5(1), pages 29-40, 1993; R. Srikant and R. Agrawal, Mining Ouantitative Association Rules In Large Relational Tables, Proceedings of the 1996 ACM SIGMOD International Conference of Management of Data, 1996; T. Fukuda, Y. Morimoto, S. Morishita, and T. Tokuyama, Mining Optimized Association Rules For Numeric Attributes, Proceedings of the 1996 ACM Symposium on Principles of Database Systems, 1996; R. Miller and Y. Yang, Association Rules Over Interval Data, Proceedings of SIGMOD '97, 1997; which are incorporated by reference herein.

Brief Summary Text (12):
Some of the techniques for knowledge extraction are for sequential patterns. R. Agrawal and R. Srikant, Mining Sequential Patterns, Proceedings of the 11th International Conference on Data Engineering, March 1995, which is incorporated by reference herein. Some of the techniques for knowledge extraction are for similarities in ordered data. R. Agrawal, C. Faloutsos, and A. Swami, Efficient Similarity Search In Sequence Databases, 4th International Conference on Foundations of Data Organization and Techniques, October 1993; C. Faloutsos, M. Ranganathan, and Y. Manolopoulos, Fast Subsequence Matching In Time-Series Databases, Proceedings of SIGMOD '94, May 1994; R. Agrawal, K. I. Lin, H. S. Sawhney, and K. Shim, Fast Similarity Search In The Presence Of Noise, Scaling And Translation In Time-Series Databases, Proceedings of the 21st International Conference on Very Large Databases, September 1995; R. Agrawal, G. Psaila, E. L. Wimmers, and M. Zait, Querying Shapes Of Histories, Proceedings of the 21st International Conference on Very Large Databases, September 1995; which are incorporated by reference herein.

Brief Summary Text (14):
The size and growth of the data stores, matched by the growing reliability and large-volume handling capability of relational database systems, has caused much of the data to be managed by these database systems. The enhancement of database systems for query optimizations and parallelization and their widening portability across a multitude of system architectures, has made the integration of data mining techniques with the database system an attractive proposition. The integration of data mining applications and database systems, however, requires appropriate data organization, some modifications and/or enhancements in the database systems, and either changes in or entirely new data mining techniques.

Brief Summary Text (15):
A very important data mining application is "association" from a database performance perspective. An association rule is a grouping of attribute value pairs. The problem of mining for association rules was introduced initially for market-basket analysis. In market-basket analysis, the association rules provided associations between the set of items purchased together in a transaction. In general, an association rule has the form A{character pullout}B, where A and B are two disjoint sets of items. The association rule conveys that the occurrence of set A in a transaction implies that the set B also occurs in the same transaction.

Brief Summary Text (17):
The term confidence is used to refer to the fraction of transactions that contain A and also contain B. Thus, support is the joint probability for A and B to occur together in a transaction, and confidence is the conditional probability for B to be found in a transaction given that A is found in it. For the generation of such rules from data mining, the user provides the minimum required support and confidence values. Then, all rules that have at least the minimum required support and confidence are generated.

Brief Summary Text (22):
In accordance with the present invention, data mining is performed by a computer to retrieve data from a data store stored on a data storage device coupled to the computer. A multi-column data store organized using a multi-column data model is received. One of the columns in the multi-column data store represents a transaction, and each of the remaining columns in the multi-column data store represents elements of that transaction. A combination operator is performed to obtain candidate itemsets of data from the multi-column data store, each itemset being a combination of a number of rows of the multi-column data store. Large itemsets of data are generated from the candidate itemsets, wherein each itemset has at least a minimum support. Association rules are generated from the large itemsets of data, wherein each association rule has at least a minimum confidence.

Drawing Description Text (3):
FIG. 2 is a flow diagram illustrating the steps performed by the Data Mining System 124;

Detailed Description Text (9):
At the center of the DB2.RTM. system is the Database Services module 114. The Database Services module 114 contains several submodules, including the Relational Database System (RDS) 116, the Data Manager 118, the Buffer Manager 120, the Data Mining System 124, and other components 122 such as an SQL compiler/interpreter. These submodules support the functions of the SQL language, i.e. definition, access control, interpretation, compilation, database retrieval, and update of user and system data. The Data Mining System 124 works in conjunction with the other submodules to perform data mining.

Detailed Description Text (15):
One embodiment of the present invention provides a Data Mining System 124. The Data Mining System 124 provides an alternate physical data model for association, referred to as a multi-column (MC) data model, that can considerably improve generation of association rules in a database. The Data Mining System 124 uses an object-relational extension of the database system (i.e., user-defined functions, or "UDFs") to generate association rules in a MC data model.

Detailed Description Text (16):
The following discussion compares the performances between the SC data model and the MC data model. One commercial implementation of association based on the Apriori technique is described in "Fast Techniques For Mining Association Rules" and supports the SC data model. The Data Mining System 124 is described as a UDF model of the Apriori technique that supports the MC data model. The experimental results indicate that there is a reduction of up to a factor of six in the execution times for the MC data model compared to the SC data model.

Detailed Description Text (17):
The Data Mining System 124 has created a new relational operator, Combinations, which is used to implement the Apriori technique's association with the MC data model. The Data Mining System 124 shows how the Combinations operator can be effectively used in SQL queries to perform the complete set of tasks in the Apriori technique. The Data Mining System 124 implements the Combinations operator using new object-relational extensions of the DB2.RTM. system for UDB for the AIX.RTM. operating system. DB2

Universal Database (UDB), http://www.software.ibm.com/data/db2/, Web Document.

Detailed Description Text (18):
Initial experiments indicate that the performance of the implementation of the Combinations operator is comparable to that of earlier pure memory based implementations of the Apriori technique. In a pure memory based implementation, all of the data for the technique is in memory for the technique. While in a database implementation of the technique, the amount of data is typically so large that the data is stored in a database, requiring a number of I/O (input/output) operations to bring data into memory from a data storage device for manipulation.

Detailed Description Text (19):
The physical data model used for data mining can have a significant impact on the performance in databases. A multi-column (MC) data model is preferable over the single column (SC) data model from a performance perspective. Enhanced object-relational extensions of the database system are used to handle the MC data model. The MC data model, though applied in one embodiment of the invention to association, could be extended, in other embodiments of the invention, to any application that requires complete scans over the data residing in database tables.

Detailed Description Text (22):
1. Determine the sets of items that have support greater than or equal to the user-specified minimum support (i.e., a support threshold). These sets of items are referred to as large itemsets. Determining large itemsets requires two elements:

Detailed Description Text (23):
a) A candidate-generation phase in which a set of itemsets, called candidate itemsets, are chosen, such that each candidate itemset contains all potential large itemsets.

Detailed Description Text (28):
This method of pruning the C.sub.I set using L.sub.I-1 results in a much more efficient support counting phase for the Apriori technique when compared to the earlier techniques. In addition, the Apriori technique uses an efficient hash-tree implementation for the candidate itemsets. This makes the process of verifying whether an itemset present in a transaction belongs in the candidate itemset or not very efficient. If the itemset belongs to the candidate itemset, its corresponding support is incremented.

Detailed Description Text (34):
For the purpose of association, the Data Mining System 124 proposes a data model of the form a transaction identifier followed by each item associated with that transaction (Transaction_id, Item_id1, Item_id2, . . . , Item_idC). The Data Mining System 124 refers to this as the multiple-column (MC) model. For example, for Transaction-1, if three items were purchased, the MC data model would show the following:

Detailed Description Text (36):
In addition to analysis of market-basket data, association has been extended to provide rules for many other forms of business data. Here, the data is quite often found organized in a schema of the form (Record_Id, Attributel, Attribute2, to AttributeC) in the MC data model. However, in order to perform association for even this kind of data, the data model is often changed to that of (Record_Id, Attribute), by pivoting the data about the values in Record_Id. The pivot operation incurs additional overhead both in terms of the CPU usage and significant disk space usage. Both these overheads are eliminated by the Data Mining System 124, which enables use of the MC data model with association rule generation.

Detailed Description Text (38):
The Data Mining System 124 proposes a new operator, Combinations, for implementing the Apriori technique through SQL on data in a MC data model. The Combinations operator takes a set of items and a number I as input, and returns the different combinations of size I from the input set of items as rows in a new table. The items in a row (i.e., the items in each itemset) are in lexicographic order. The result table has I number of columns. For example, for a row in the input table that contains the items A, B, C and D, the output of the Combinations operator, invoked with I=2, would be (A,B), (A,C), (A,D), (B,C), (B,D) and (C,D). In one embodiment, the Combinations operator could be implemented as a table function of the DB2.RTM. system for UDB for the AIX.RTM. operating system.

Detailed Description Text (63):

The Data Mining System 124 implements the Combinations operator using the object--relational extension, table--function. Like a UDF, the table--function is a user--defined method that can be used in SQL queries. The table--function returns a table to the query invoking it and is used in the FROM clause of an SQL query. This extension provides a highly flexible table construction utility to be used along with SQL.

Detailed Description Text (65):
The Data Mining System 124 again uses a modified form of the Combinations operator to implement the association rule-generation with SQL. The modified operator, Combinations_plus, returns an input value, the support count, and the rest of the items in the input set, along with each of the combinations in the input set. The Combinations_plus operator can be used to generate the association rules as follows:

Detailed Description Text (75):
FIG. 2 is a flow diagram illustrating the steps performed by the Data Mining System 124. In Block 200, the Data Mining System 124 receives a multi-column data store organized using a multi-column data model. One of the columns in the multi-column data store represents a transaction, and each of the remaining columns in the multi-column data store represents elements of that transaction. In Block 202, the Data Mining System 124 performs a combination operator to obtain candidate itemsets of data from the multi-column data store, each itemset being a combination of a number of rows of the multi-column data store. In Block 204, the Data Mining System 124 generates large itemsets of data from the candidate itemsets, wherein each itemset has at least a minimum support. In Block 206, the Data Mining System 124 generates association rules from the large itemsets of data, wherein each association rule has at least a minimum confidence.

Detailed Description Text (77):
The following discussion provides an example to better illustrate the Data Mining System 124. The Data Table provides input data, and the L.sub.1 Table provides large itemsets of size 1 and their counts. The minimum support is two (i.e., 50%) in this example, and the minimum is 0.5.

Detailed Description Text (105):
In the large-itemset generation phase, using the Combinations operator on all of the items in one row (i.e., all of the items in a transaction), could result in too many unwanted combinations being generated. These would be eliminated by the Join with the candidates table C.sub.I. However, performance could be greatly enhanced if they were not even generated. In order to do this, the Data Mining System 124 adds a new clause to the Combinations operator, an "in Set" clause. With the "in Set" clause, the Combinations operator generates combinations of only those items in the input set which are present in the set identified by the "in Set" clause. Incorporating the "in Set" clause into a table-function is done by passing the "in Set" clause through a pointer to a hash table containing all of the items in C.sub.I. This is a relatively small set, and the memory management for this is trivial, compared to maintaining all of the candidate sets in memory.

Detailed Description Text (106):
In the candidates generation phase, the Data Mining System 124 can again use L.sub.I-items (i.e., a subset of the set of items in L.sub.I) in place of L.sub.I (i.e., the set of items in the large itemset). L.sub.I-items is a subset of L.sub.I that becomes much smaller as I increases. Using a subset of the large itemset reduces the number of invocations of the Combinations operator for determining the candidates itemset.

Detailed Description Text (108):
An optimization of the Data Mining System 124 can approximate the performance of that of UDF (in-memory) implementation of the Apriori technique. Currently in the execution, when generating the large itemset table, the output of the Combinations operator is created as a temporary table in the database and then joined with the candidates table. The Data Mining System 124, as optimized, avoids generating this temporary table, but directly increments the counts for the candidate sets (i.e., entries in the candidates table) when the table function outputs a suitable row (i.e., one that has a corresponding candidate table entry), and, thus, enhances the performance significantly.

Detailed Description Text (109):
Meo et. al. had proposed a SQL-like operator for mining association rules. R. Meo, G.

Psaila, and S. Ceri, A New SQL-Like Operator For Mining Association Rules, Proceedings of the 22nd International Conference on Very Large Databases, 1996, which is incorporated by reference herein. Their work focused on providing a unifying model for the description of association rules, as opposed to an implementation of association using the database engine that the Data Mining System 124 provides.

Detailed Description Text (111):
Comparison Of The MC Data Model And The SC Data Model

Detailed Description Text (112):
The Data Mining System 124 uses a simplified table-scan cost model to illustrate the benefit of the MC data model. Consider a table with R Transaction_ids, each one involving C number of items on the average. For the CPU costs for accessing the input data table, let t.sub.R denote the time associated with a row access for concurrency control, page-fix, etc. and let t.sub.c denote the CPU time required for a single column access. Then the CPU costs associated with both the models are as follows:

Detailed Description Text (114):
Thus, the Data Mining System 124 shows that the CPU cost difference between the SC data model and the MC data model is equal to the (number of transactions) times the (time for row access) times (one less than the number of items). Thus, the CPU cost difference increases as the number of rows in the table increases and also as the average number of items in a transaction grows. That is, the SC data model gets more expensive based on these factors.

Detailed Description Text (117):
The volume difference is the (number of transaction identifiers) times ((the size of a header of a row) times (one less than the number of items in a transaction) plus (the size of the transaction identifier times the number of items). Here, the Data Mining System 124 shows that the I/O cost difference also increases with an increase in the number of rows and with an increase in the average number of items in a transaction.

Detailed Description Text (118):
In the above models, the cost for handling the null entries in the MC data model was ignored. If t.sub.n is considered the CPU cost for processing a null item, the Data Mining System 124 obtains the condition (N-C)/(C-1) i t.sub.R /t.sub.n for the SC data model to have a higher CPU cost. Similarly, if S.sub.n is considered to be the size of the null indicator, for the I/O cost of the SC data model to be greater than the I/O cost of the MC data model, the Data Mining System 124 gets the condition (S.sub.h *(C-1)+S.sub.t *C)>S.sub.n *N. As long as the value of N is appropriately chosen (relative to C), these conditions would hold for most databases.

Detailed Description Text (119):
The performance of two implementations for association are compared, one working with the SC data model and the other with the MC data model for the input database table. Both the implementations are based on the Apriori technique. The Apriori technique was chosen because it does not require the use of relational operations in its implementation, and, thus, can operate without the SC data model. Also, the Apriori technique is a widely used point of comparison for various studies on association. In addition, the Apriori technique is the technique of choice for the commercial implementation of association in IBM's Intelligent Miner (IM) Data Mining Suite. IBM Intelligent Miner, http//www.software.ibm.com/data/intelli-mine/, Web Document.

Detailed Description Text (120):
The Data Mining System 124 modifies the optimized Intelligent Miner implementation of the Apriori technique for the SC data model. For the MC data model, the Data Mining System 124 uses one of the object-relational extensions for the DB2.RTM. system, called user-defined functions (UDFs), which are further described in D. Chamberlin, Using The New DB2-IBM's Object-Relational Database System, Morgan Kaufmann, 1996, which is incorporated by reference herein. A UDF is a user-defined method that provides an application program the means to perform computation on retrieved records inside the database system. In one implementation, every row of data is passed from the MC data model table to the UDF function. All of the computations and generation of rules is done by the code inside the UDF.

Detailed Description Text (121):
The UDFs in the DB2.RTM. system can be run in two modes, either fenced or unfenced. In the unfenced mode, the UDFs share the database's address space, avoiding the overhead of switching from the database's address space to the application's address space, when

data is obtained from the database to the application. In the fenced mode, the UDFs run in the application's address space, which is distinct from the database's address space. Agrawal and Shim have shown the benefit to be obtained by running UDFs in the unfenced mode. R. Agrawal and K. Shim, Developing Tightly-Coupled Data Mining Applications On A Relational Database System, Proceedings of the 2nd International Conference on Knowledge Discovery in Databases and Data Mining, August 1996, [hereinafter "Developing Tightly-Coupled Data Mining Applications On A Relational Database System"], which is incorporated by reference herein. Experiments were conducted with UDFs running in the fenced mode, to provide the same environment for the current implementation as for the Intelligent Miner implementation of the Apriori technique, in which all of the computations are performed in the application's address space.

Detailed Description Text (122):
Agrawal and Shim, in "Developing Tightly-Coupled Data Mining Applications On A Relational Database System", have shown the benefit of using UDFs for the development of applications tightly coupled with the database engine. Their focus in using the UDF for association, however, was not for its use with the MC data model.

Detailed Description Text (123):
Tests were also performed with a UDF implementation for the SC data model, and the UDF implementation for the SC data model was up to a factor of 2-3 times slower than the highly optimized Intelligent Miner implementation of the SC data model. Hence, the Data Mining System 124 uses Intelligent Miner's implementation as representative of the SC data model. The data structures and the computation modules are quite similar for both the Intelligent Miner and UDF implementations. The only difference lies in the extraction of data from the database. The Intelligent Miner obtains the data from a table with data in the SC data model, and the UDF obtains the data from a table in the MC data model.

Detailed Description Text (124):
The experiments were conducted on an IBM PowerStation 590, that has a 66 Mhz Power2 processor with 512 MB memory, with a 1.07 GB serial-link disk drive. The mining data was drawn from sales data of a retail store chain, with transactions drawn over various periods of time. The data has an average of 12 items per sale (i.e., the SC data model has about 12 times the number of rows as the MC data model). The UDF implementation for the MC data model supported a maximum of 60 items per transaction, and the input data under both the SC and MC data models were identical.

Detailed Description Text (129):
The Data Mining System 124 has shown that the MC data model is the better physical data model, as compared to the SC data model, for association. The results show that significant performance improvement can be obtained for the MC data model. The Data Mining System 124 has also provided the operator ("Combinations") necessary for performing SQL-query based implementation of the Apriori technique over the MC data model. The initial experiments with the implementation of the Combinations operator using the table-function object-relational extension yields performance comparable to that of Intelligent Miner. This provides a convenient method for performing association in the database engine using the optimization and parallelization in the database system for the benefit of the application. It also provides a scalable implementation, which is not memory bound, for the Apriori technique.

Detailed Description Text (130):
The Data Mining System 124 provides careful attention to the data model and access pattern and provides suitable extensions to the relational database system to meet the new requirements of data mining applications.

Current US Original Classification (1):
707/2

CLAIMS:

1. A method of data mining in a computer, the data mining being performed by the computer to retrieve data from a data store stored on a data storage device coupled to the computer, the method comprising the steps of.

receiving a multi-column data store organized using a multi-column data model, wherein one of the columns in the multi-column data store represents a transaction and each of the remaining columns in the multi-column data store represents items of that

transaction;

performing a combination operator in a relational database management system to obtain candidate itemsets of data from the multi-column data store, each itemset being a combination of a number of rows of the multi-column data store;

generating large itemsets of data from the candidate itemsets, wherein each itemset has at least a minimum support; and

generating association rules from the large itemset of data, wherein each association rule has at least a minimum confidence.

8. The method of claim 1, wherein confidence is a conditional probability for a first element to be found in a transaction given that a second element is found in the transaction.

9. An apparatus for data mining, comprising:

a computer having a memory and a data storage device coupled thereto, wherein the data storage device stores a data store;

one or more computer programs, performed by the computer, for receiving a multi-column data store organized using a multi-column data model, wherein one of the columns in the multi-column data store represents a transaction and each of the remaining columns in the multi-column data store represents items of that transaction, for performing a combination operator in a relational database management system to obtain candidate itemsets of data from the multi-column data store, each itemset being a combination of a number of rows of the multi-column data store, for generating large itemsets of data from the candidate itemsets, wherein each itemset has at least a minimum support, and for generating association rules from the large itemset of data, wherein each association rule has at least a minimum confidence.

16. The apparatus of claim 9, wherein confidence is a conditional probability for a first element to be found in a transaction given that a second element is found in the transaction.

17. An article of manufacture comprising a program storage medium readable by a computer and embodying one or more instructions executable by the computer to perform method steps for data mining, the data mining being performed by the computer to retrieve data from a data store stored on a data storage device coupled to the computer, the method comprising the steps of:

receiving a multi-column data store organized using a multi-column data model, wherein one of the columns in the multi-column data store represents a transaction and each of the remaining columns in the multi-column data store represents items of that transaction;

performing a combination operator in a relational database management system to obtain candidate itemsets of data from the multi-column data store, each itemset being a combination of a number of rows of the multi-column data store;

generating large itemsets of data from the candidate itemsets, wherein each itemset has at least a minimum support; and

generating association rules from the large itemset of data, wherein each association rule has at least a minimum confidence.

24. The article of manufacture of claim 17, wherein confidence is a conditional probability for a first element to be found in a transaction given that a second element is found in the transaction.

WEST

Generate Collection Print

L16: Entry 3 of 9 File: USPT Oct 30, 2001

DOCUMENT-IDENTIFIER: US 6311173 B1
TITLE: Pattern recognition using generalized association rules

Brief Summary Text (2):
The present invention relates generally to systems and methods for pattern recognition, and specifically to methods of pattern recognition based on association rules.

Brief Summary Text (4):
Automated pattern recognition is well known in the art, in a variety of applications. Common applications of pattern recognition include image analysis, speech recognition, and predicting unknown fields for records in a database. Typically, a template or a collection of rules is determined, which are believed to constitute a pattern that is characteristic of a certain class. Items in the set are then evaluated to determine how closely they fit the pattern. A close fit indicates a high probability that the item being evaluated falls within the class. Thus, a face may be found to belong to a certain individual; or a spoken sound may be found to correspond to a certain word; or a bank customer may be predicted to be a good or bad credit risk.

Brief Summary Text (5):
In order to build the template or rules, "data mining" of a training database is frequently used. The training database is selected and is assumed to be a real and representative sample of the overall population. The training database generally contains variables (fields, representing various attributes of the items) of different types, among which a field is selected as the "Field to Predict" (also referred to in the database art as the "output", or "result", or "dependent" variable). The training database may be represented as a temporary file of codes of attribute values, given by the matrix:

Brief Summary Text (8):
The accuracy of prediction of the value of y may be verified by testing on the training database. However, such testing may not be sufficient, since there exists the problem of "overfitting," wherein regularities discovered on the training database turn out to be found by chance, and are therefore not valid on other samples from the overall population. Thus, the purpose of data mining is to discover regularities within the training database which possess the property of likely stability of their validity on the whole population.

Brief Summary Text (9):
Regularities of this sort are sought within the training database in the form of association rules. Methods of deriving such association rules are described, for example, by Agrawal, et al., in "Fast Discovery of Association Rules," in Advances in Kowledge Discovery and Data Mining (AAAI Press/MIT Press, 1996), pages 307-328; and by Zaki, et al., in "New Algorithms for Fast Discovery of Association Rules," in Proc. 3rd Int. Conf. KDD (California), pages 283-286. These publications are incorporated herein by reference.

Brief Summary Text (17):
Which rules are interesting? In other words, about which rules can it be said that they were discovered not by chance, and are likely to be valid on the overall population? It can be assumed that these are rules fulfilled at a sufficiently large number of records and whose probability significantly deviates from p.sub.a. A formal statement for this intuitive notion is as follows: A user specifies a minimum support S.sub.min, and minimum admissible probabilities for a rule with y=1 (denoted by p.sub.1) and with y=0 (denoted by p.sub.0), 1-p.sub.0 <p.sub.a <p.sub.1. The objective of the data mining is then to determine association rules (rules of the type of equation (2)) for which s.gtoreq.S.sub.min and p.gtoreq.p.sub.1 (if y=1), or p.gtoreq.p.sub.0 (if y=0). Methods

of data mining known in the art do not necessarily find all such rules exhaustively on the training database, and furthermore tend to require very substantial computing resources.

Brief Summary Text (18):
The rule defined by equation (3) can be expressed as a statement of conditional probability:

Brief Summary Text (19):
However, unlike equation (3), the number of records at which this statement is fulfilled (support s) is absent in equation (4). Therefore, association rules in a sense contain more information than conditional probabilities, which are applied in Bayes methods. Such methods are described, for example, by Friedman, in "On Bias, Variance, 0/1--Loss, and the Curse-of-Dimensionality," in Data Mining and Knowledge Discovery, 1(1), pages 54-77; and by Heckerman, in "Bayesian Networks for Data Mining," in Data Mining and Knowledge Discovery, 1(1), pages 79-119. These publications are incorporated herein by reference.

Brief Summary Text (23):
It is an object of the present invention to provide improved systems and methods of pattern recognition. In some aspects of the present invention, these systems and methods are used for predicting an attribute of one or more items in a population.

Brief Summary Text (27):
The term "generalized association rules," as used in the present patent application and in the claims, refers to rules that use logical operations of conjunction, disjunction, and negation in defining their conditions. As described above, methods of data mining and prediction known in the art use only simple association rules, based only on the logical conjunction operation. Methods in accordance with the present invention, using generalized association rules, provide "stronger" and more stable rules, which afford more accurate and reliable prediction, at lower expenditure of computing resources, than methods known in the art. Furthermore, the present invention substantially overcomes the problem of overfitting, which exists in methods known in the art.

Brief Summary Text (28):
In some preferred embodiments of the present invention, before determining, the generalized association rules, substantially all of the simple association rules meeting the minimum support and minimum probability criteria are found, using generalized contingency tables and sets of potentially representative q-conditions for each possible value of q. Methods of data mining known in the art do not use contingency tables and are not capable of conclusively finding all simple association rules. Because methods in accordance with the present invention find an exhaustive set of rules, they allow the attributes of the items in the database to be predicted with the highest possible level of confidence.

Brief Summary Text (30):
Preferred embodiments of the present invention are described herein with reference to methods and systems for data mining and prediction of unknown fields in a database. It will be appreciated, however, that the principles of the present invention may similarly be applied in other areas of pattern recognition. For example, generalized association rules may be derived for use in image or voice recognition, and may thus be used to identify with improved accuracy, reliability, and computation speed the identity of an item in the image or the word associated with a spoken sound pattern.

Brief Summary Text (60):
In a preferred embodiment, a probability decision point is determined such that when the cumulative probability is greater than the decision point, the attribute of interest is predicted to have a first value, and when the probability of interest is less than the decision point, the attribute of interest is predicted to have a different, second value. Preferably, determining the decision point includes defining an ambiguity range of probabilities including the decision point in which the predicted value is ambiguous, most preferably by comparing the training sample and at least a portion of the overall population from which the given item is taken, and determining an extent of the ambiguity range responsive to a measure of the similarity of the training sample and the at least portion of the overall population.

Brief Summary Text (76):
In a preferred embodiment, the processor determines a probability decision point such that when the cumulative probability is greater than the decision point, the attribute

of interest is predicted to have a first value, and when the probability of interest is less than the decision point, the attribute of interest is predicted to have a different, second value. Preferably, the processor defines an ambiguity range of probabilities including the decision point in which the predicted value is ambiguous. Most preferably, the processor compares the training sample and at least a portion of the overall population from which the given item is taken, and determines an extent of the ambiguity range responsive to a measure of the similarity of the training sample and the at least portion of the overall population.

Drawing Description Text (2):
FIG. 1 is a flow chart illustrating a generalized method for data prediction based on pattern recognition, in accordance with a preferred embodiment of the present invention;

Detailed Description Text (3):
As shown in FIG. 2, system 20 comprises an input device 22, which receives data relating to a plurality of items in a population and conveys the data to a processor 24, preferably a general purpose computer. Device 22 preferably comprises a keyboard or digital data link, through which database records regarding the items in the population are input to the processor. Alternatively, device 22 may comprise an electronic camera or scanner, for inputting image data; or a microphone, for inputting audio data; or any other suitable type of sensor or other input device known in the art. Although preferred embodiments are described hereinbelow mainly with reference to analysis and prediction of database records, in alternative embodiments of the present invention, system 20 and the method of FIG. 1 may be applied to pattern recognition and prediction on other types of data, as well.

Detailed Description Text (14):
Each individual field of the training database is considered to be a qualitative (categorical) or quantitative variable. The variable type is defined by the types of comparisons that can be applied to its values. If only the relations "equal" and "unequal" can be introduced on the set of a variable's values, then such a variable is said to be a qualitative (categorical) variable. Unlike such qualitative variables, the set of values of a quantitative variable has the structure and properties of the axis of real numbers. All alphanumeric data are considered as attributes. Therefore, an attribute can be recognized by definition of the source file when the field is defined as alphanumeric. However, if the field is defined as numeric, it is impossible to ascertain whether the variable is quantitative or qualitative. In this case, the user must state the type of variable corresponding to this field. In the present patent application, we will not distinguish between a categorical and an orderable qualitative variable, wherein an ordered relation between any pair of variable values can be introduced. All such variables are collectively referred to herein as "attributes."

Detailed Description Text (124):
In methods of pattern recognition and prediction known in the art, prediction of unknown values is carried out on the basis of simple association rules determined on the training database. The inventors have discovered that generalized association rules, which are found as described herein, enable unknown fields to be predicted with greater confidence than is generally possible with simple rules alone. However, the techniques of coding variables and determining simple association rules in accordance with the principles of the present invention, as described hereinabove, may also be used advantageously in prediction using simple association rules.

Detailed Description Text (139):
Assume that we have determined a serial number k.sub.b such that y=1 is predicted for all records corresponding to k>k.sub.b, and y=0 is predicted for all records corresponding to k.ltoreq.k.sub.b. In the example of FIG. 9, if k.sub.b =4, then the vector of predicted values of y consists of three ones followed by 17 zeroes. Comparing this Boolean vector with the Boolean vector of actual values of y in the last line of the table, the number of non-coincident values in corresponding components of these two vectors (equal to the number of ones in the modulo-2 sum of these vectors) will be equal to the number of errors in the prediction. For k.sub.b =4, the number of errors in the prediction is equal to 3 (at columns 4, 5 and 7).

Detailed Description Text (189):
In some preferred embodiments of the present invention, the methods described above are used to output predictions as to one or more attributes of items whose characteristics are stored in respective records in the database, for example, so as to sort the items into groups according to an attribute of interest. In other preferred embodiments,

variants of these methods are used to identify and/or sort input images or sounds according to patterns appearing in appropriate sensor data received from input device 22 (FIG. 2). Although certain specific preferred embodiments are described hereinabove with reference to database applications, those skilled in the art will appreciate the wide range of applications of and variations on the methodology described herein, all of which are within the scope of the present invention.

Current US Cross Reference Classification (2):
707/1

Current US Cross Reference Classification (3):
707/6

Other Reference Publication (1):
Agrawal et al, "Mining Association Rules Between Sets of Items in Large Databases", Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data, May 1993.*

Other Reference Publication (5):
Agrawal, et al., "Fast Discovery of Association Rules", in Advances in Knowledge Discovery and Data Mining (AAAI Press/MIT Press, 1996), pp. 307-328.

Other Reference Publication (7):
Friedman, "On Bias, Variance, 0/1--Loss and the Curse-of-Dimensionality", in Data Mining and Knowledge Discovery, 1(1), pp. 54-77.

Other Reference Publication (8):
Heckerman, "Bayesian Networks for Data Mining", in Data Mining and Knowledge Discovery, 1(1), pp. 79-119.

CLAIMS:

29. A method according to claim 28, wherein defining the ambiguity range comprises comparing the training sample and at least a portion of the overall population from which the given item is taken, and determining an extent of the ambiguity range responsive to a measure of the similarity of the training sample and the at least portion of the overall population.

61. A system according to claim 60, wherein the processor compares the training sample and at least a portion of the overall population from which the given item is taken, and determines an extent of the ambiguity range responsive to a measure of the similarity of the training sample and the at least portion of the overall population.